Starting cPlay

When you run cPlay (double-click desktop icon) for the first time, setup must be performed.

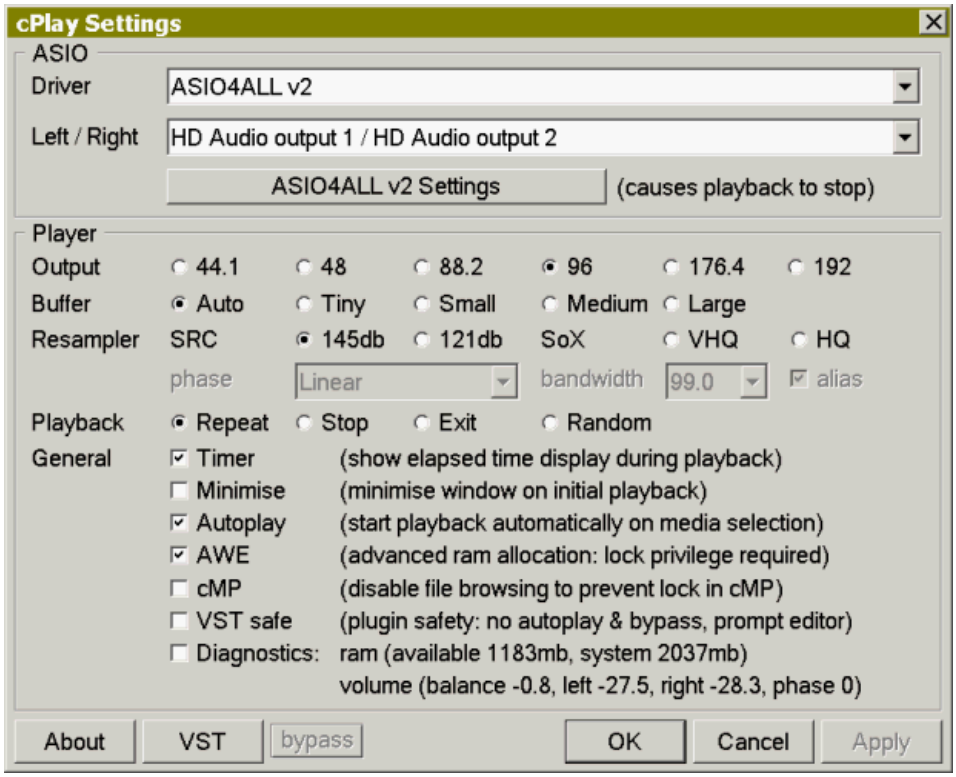
Playing audio with RealTime priority:

1. Select .cue (playlist), .wav or .flac file from File Explorer
2. Select 'Open With' (right-click on file to be played)
3. Locate and select 'cicsPlay.bat' where cPlay was installed (default is 'c:\program files\cics Play'). Note: If you did NOT use the default install folder, edit cicsPlay.bat and correct the path. Hint: use command 'dir *.* /x' to identify folder's DOS shortname.

Using Cue Sheets which enables playlist playback is recommended.

Settings

Bottom left button in main screen brings up cPlay's settings. ASIO settings button below depends on ASIO driver – Juli@ has no effect whilst RME, Lynx, EMU and ASIO4ALL gives access to ASIO control panel of driver. cPlay offers 2 channel output mapping only (this is a deliberate design choice).



For some changes to take effect (ASIO, Buffer, AWE and Output Rate), a restart is required. ASIO changes (from cPlay or external) will cause cPlay to stop (and a restart is required). This is an important safety measure.

When changing rates (e.g. 96 to 44.1), some ASIO soundcards will re-map its channels – cPlay tries to detect this (depending on ASIO driver) and informs you to reset left & right channels. Why? Soundcards using the ADAT interface use channel multiplexing (i.e. 96k gives 2x48k...) hence channels change with rate.

cMP setting disables file browse button (to prevent indefinite waits in cMP²).

Buffer

"DSP Buffer" at Auto (default), Small, Medium or Large is provided.

Guidance: CPUs with 2MB+ L2 cache, use Auto setting. Small is ideal for high output rates (above 96k) whilst Medium is best for 96k or lower. On CPUs offering less than 2MB L2 cache, Small is recommended for all output rates. 'Large' buffer setting is used by Auto when output rate is 48k or less. Processors with L2 cache size larger than 2MB, its worth testing Medium and Large settings (for any output rate).

"Tiny" buffer option offers lowest L2/L3 cache footprint enabling lower specification CPUs to be used with resampling. The 2MB L2/L3 cache requirement remains to produce highest quality 192k output. Auto only uses Small, Medium, or Large, i.e. Tiny must be manually selected. Tiny is highly recommended when using a VST plugin.

These buffer sizes work best with low ASIO latency setting (less than 512 samples).

Why buffer sizes matters?

Buffer setting affects resource consumption of CPU and RAM. This in turn affects power supply and the ground plane. Another key factor is how often the DSP thread is used to fill buffers. It's a periodic transaction and hence causes periodic jitter. Every player has this. Larger buffers require longer CPU bursts (especially so with SRC@145db). When using cMP, larger buffers cause slower UI responses, for example one gets a "slow" or "lazy" mouse with larger buffers. This is entirely by design in cMP² with Optimise set to "Critical". Try setting cMP's Optimise to "Player" or "Realtime" and you'll experience different results.

> *Explanation:* when you move the mouse, it requires CPU intervention but with a large DSP load there's a delay as the DSP thread enjoys RealTime priority in cMP. When Optimise is not set to "Critical", the mouse action is serviced immediately by the other (most likely) free CPU core (when using a dual core CPU). This unfortunately interferes with our most critical ASIO thread: an optimal design has a CPU Core dedicated to it as this not only circumvents but is superior to having a RealTime OS kernel.

Therefore, different buffer sizes induce (by software) different jitter distortion (both in magnitude and frequency) causing audible differences.

SRC @145.68db SNR

cPlay can use the CPU intensive 145.68db SNR upsampler. A word of caution: do NOT attempt to use this on lightweight processors (anything less than E4xxx Core 2 Duo processor). **A minimum of 2MB L2 cache is required.** For example, the E2140 (1MB L2 cache) and Pentium 4 3GHz (1MB L2 cache) can only accomplish 44.1->96 - anything beyond this results in a locked/slow computer. There's a tradeoff here: more CPU power means more electrical inteference (and increased power consumption) thus reducing benefits of superior upsampling.

The E6300 (2MB L2 cache, 1.86GHz) processor comfortably copes with 192k upsampling with CPU load at ~40% (individual CPU core). cMP² using this setup operates with no CPU fan gives CPU temperatures of ~43°C. In cMP², no dropouts occur when Optimize is set to Critical. Intel's new E7xxx (based on 45nm technology) is an excellent choice offering low power consumption and brutal performance.

SoX Resampler Settings

SoX implementation in cPlay enjoys full optimisation as that for SRC (Secret Rabbit Code). This includes 128bit DSP processing and other optimisations. Only VHQ (Very High Quality, 175db rejection aka Stop Band) and HQ (High Quality, 125db rejection) converters are supported. SRC at 145db SNR (154db rejection) and 121db SNR (120db rejection) remains as before. SoX VHQ offers better than 170db SNR performance (see measurements below). Other SoX resampler options available in cPlay are (as per SoX Manual):

- Phase (0-100)

  All resamplers use filters that can sometimes create `echo' (a.k.a. `ringing') artefacts with transient signals such as those that occur with `finger snaps' or other highly percussive sounds. Such artefacts are much more noticable to the human ear if they occur before the transient (`pre-echo') than if they occur after it (`post-echo'). Note that frequency of any such artefacts is related to the smaller of the original and new sampling rates but that if this is at least 44.1kHz, then the artefacts will lie outside the range of human hearing.

  A phase response setting may be used to control the distribution of any transient echo between `pre' and `post': with minimum phase (0), there is no pre-echo but the longest post-echo; with linear phase (50), pre and post echo are in equal amounts (in signal terms, but not audibility terms); the intermediate phase (25) setting attempts to find the best compromise by selecting a small length (and level) of pre-echo and a medium lengthed post-echo. Note that phase responses between `linear' (50) and `maximum' (50..100) are rarely useful.

- Bandwidth (90-99.7%)

  Band-width is the percentage of the audio frequency band that is preserved. A resampler's band-width setting determines how much of the frequency content of the original signal (w.r.t. the orignal sample rate when up-sampling, or the new sample rate when down-sampling) is preserved during conversion. The term `pass-band' is used to refer to all frequencies up to the band-width point (e.g. for 44.1kHz sampling rate, and a resampling band-width of 95%, the pass-band represents frequencies from 0Hz (D.C.) to circa 21kHz). Increasing the resampler's band-width results in a slower conversion and can increase transient echo

artefacts (and vice versa).

The -s `steep filter' option (99.0% bandwidth) changes resampling band-width from the default 95% (based on the 3dB point), to 99%. Band-width values greater than 99% are not recommended for normal use as they can cause excessive transient echo.

- (Above Bandwidth) Aliasing

  Aliasing above the pass-band is allowed. For example, with 44.1kHz sampling rate, and a resampling band-width of 95%, this means that frequency content above 21kHz can be distorted; however, since this is above the pass-band (i.e. above the highest frequency of interest/audibility), this may not be a problem. The benefits of allowing aliasing are reduced processing time, and reduced (by almost half) transient echo artefacts.

Using AWE

Physical RAM allocation method is used (for WAV RAM LOAD). This advanced technique requires LOCK privilege setting! AWE stands for Address Windowing Extensions - allocation occurs directly (system available RAM drops immediately). Windows may NOT allocate all 'Available RAM' - in which case, cPlay reverts to standard approach

Diagnostics will show whether AWE was successful at RAM allocation. Process Explorer or Task Manager will not show RAM allocated to cPlay's working storage, instead you'll see the reduction in Available RAM.

Make sure you have LOCK privileges set on your computer. Windows will not offer all "Available RAM" for AWE - in which case cPlay reverts to standard allocation. Adding additional RAM will prevent this. AWE offers the potential to load up to 4GB of RAM - not tested.

How 2 set LOCK privilege

Perform these steps exactly (applicable to XP SP2 Pro):

1. Start Menu > Run > enter "mmc" > OK button
2. File > Add/Remove Snap-in > Add button
3. Select "Group Policy Object Editor" > Add button
4. Finish button
5. Close button
6. OK button
7. Local Computer Policy > Computer Configuration > Windows Settings > Security Settings > Local Policies > select "User Rights Assignment"
8. Double-click "Lock pages in memory"
9. Add User or Group button
10. Object Types... button
11. check Groups > uncheck Built-in security principals & Users > OK button
12. Enter "Administrators" under "Enter the object names to select" > OK button
13. Apply button > OK button
14. File > Save
15. File > Exit
16. Reboot

Lock privilege is now enabled. If this is not done, cPlay will fail on AWE and revert to standard RAM allocation. This is reported in Diagnostics only, i.e. "AWE allocation successful" message is given.

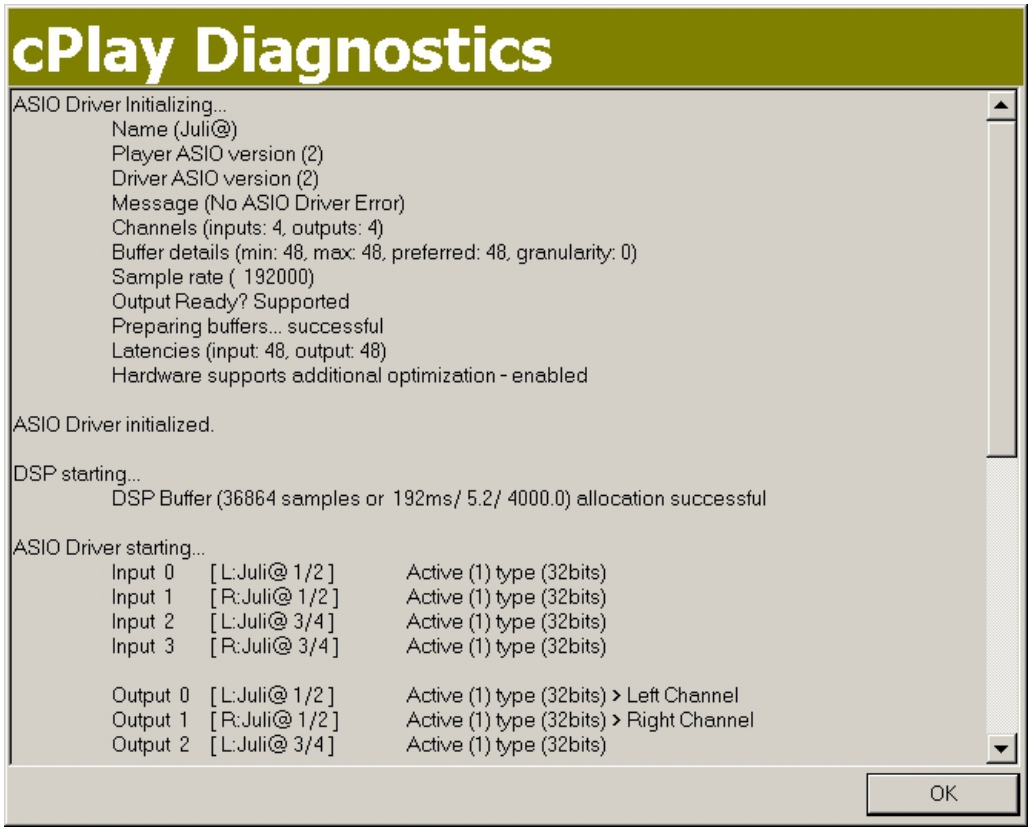
Keyboard Actions

- '[' / ']' for Previous / Next
- '-' / '=' for Skip Backwards / Skip Forwards
- ';' for phase 0/180
- '.' for stop
- 'p' for play / pause
- shift+Home for volume up
- shift+End for volume down
- Enter to play highlighted track
- Alt+F4 or Esc for exit

Diagnostics

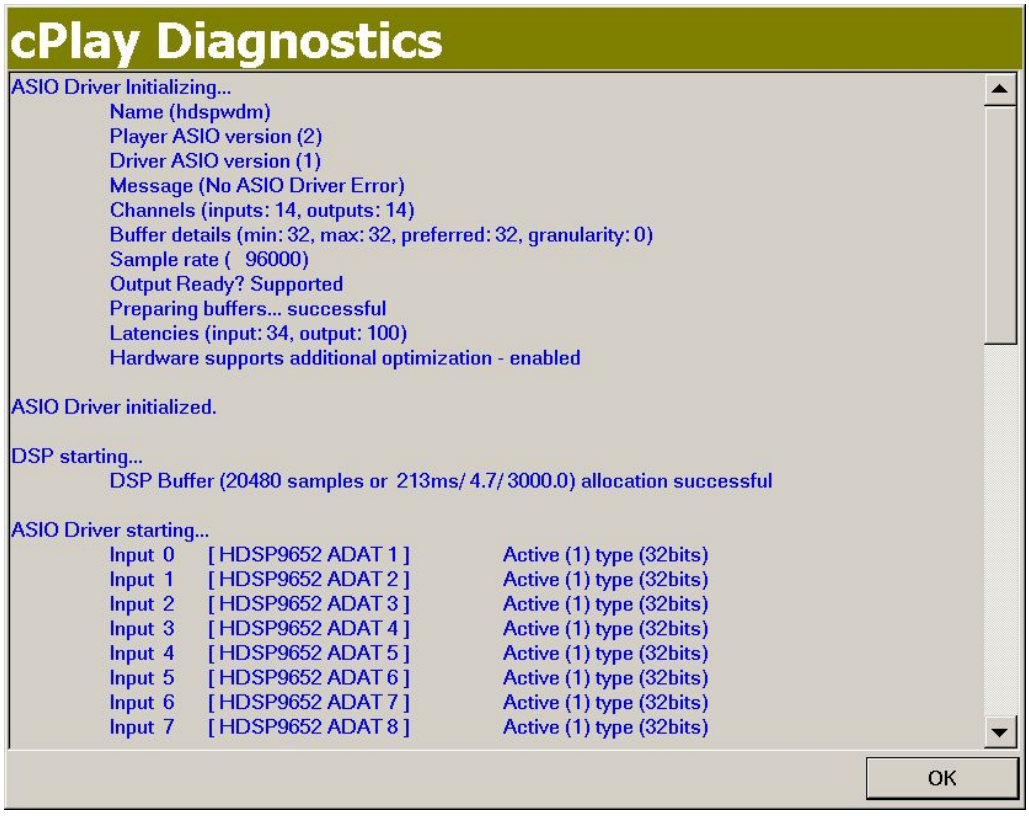
Incredible detail is provided when diagnostics is activated. Also, when cPlay encounters an error, diagnostics is displayed – it's always a good idea to scroll to the bottom for important messages.

Here's the initial startup diagnostics using Juli@:



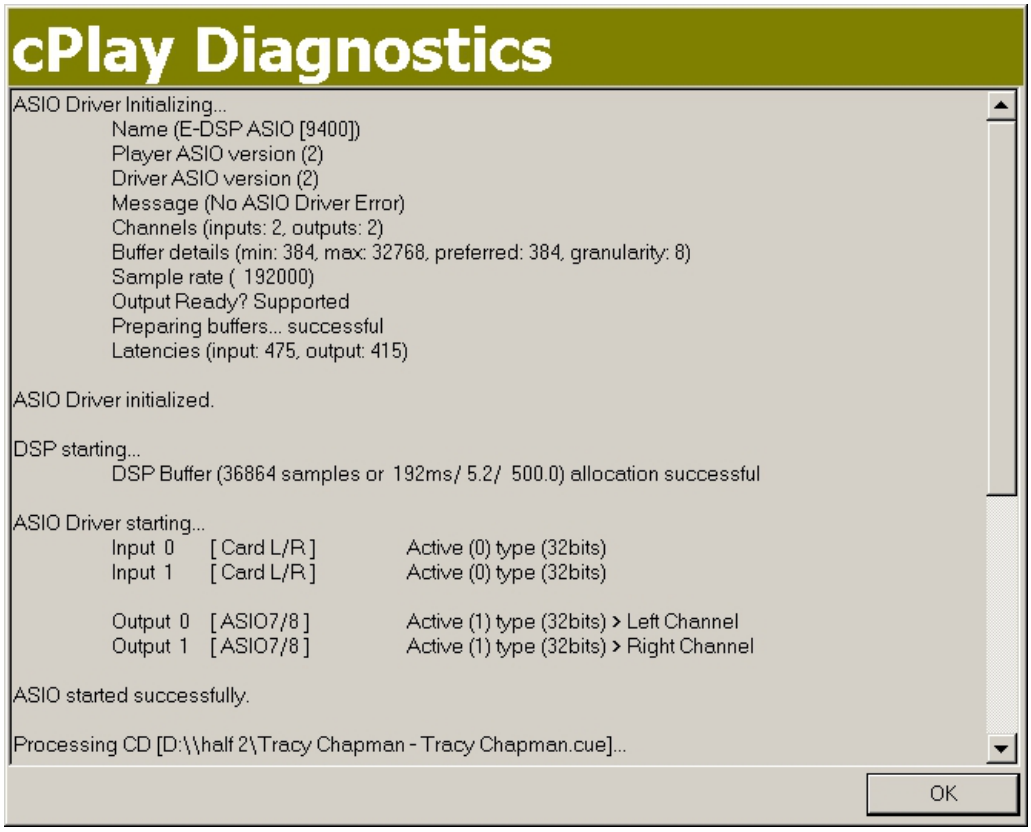
Both Player (cPlay) and Driver (Juli@) support ASIO version 2.0. Latency at 'preferred buffer' level matches output latency (48 samples), output ready & hardware optimizations are done. Any advantage of RME's off-host processing is negated here. Juli@ passes with flying colors!

For RME HDSP9652:



Disappointing to see only support for ASIO 1.0 and actual output latency differs from preferred buffer latency - it's worse than Juli@. Ideally, both output and buffer latencies should be the same. Output ready & hardware optimizations are done.

For EMU 1212M:



This soundcard offers poor latency of 2ms (at any sample rate), preferred buffer latency doesn't match output latency and no hardware optimization supported. Note that if you have an EMU card installed, do NOT disable it (in device manager) as this causes instability (as drivers remain operational)!

VST Plugin

VST is Steinberg's standard for applying DSP sound effects to audio streams. Effects can be absolutely anything one desires (stereo to mono, EQs, Stereo to 5.1., etc.). There's a huge selection - one needs to be careful as some have bad quality (bugs and/or process in integer causing quality loss). Used correctly, one could achieve excellent results. Another common acronym is VSTi which are DSP components that act like musical instruments.

Here's Steinberg's explanation of a VST plugin:

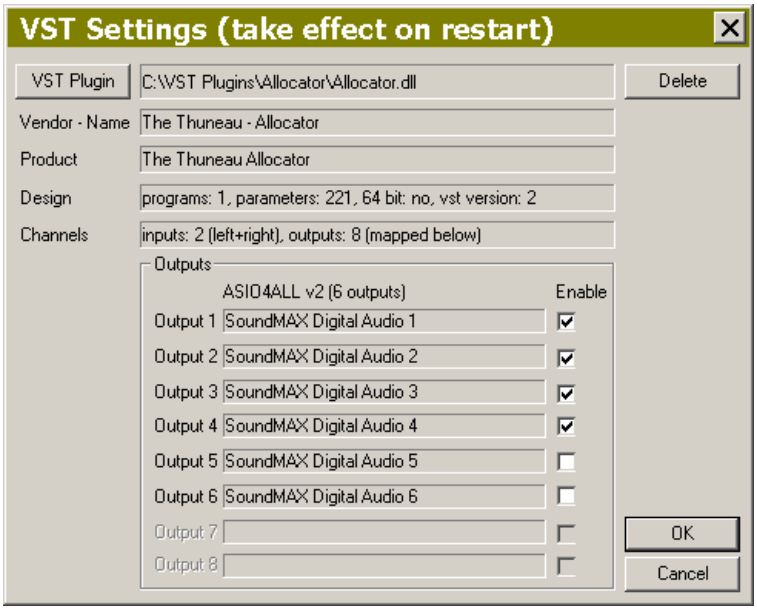
> Essentially, a VST Plug-in is a pure audio processing component, and not an audio application: It is a component that is utilized within a host application. This host application provides the audio streams that are processed by the plug-in's code.
>
> Generally speaking, a VST plug-in it can take a stream of audio data, apply a process to the audio, and return the result to the host application. A VST Plug-In performs its process normally using the processor of the computer; It does not necessarily need dedicated digital signal processors. The audio stream is broken down into a series of blocks. The host supplies the blocks in sequence. The host and its current environment control the block-size. The VST Plug-In maintains the status of all its own parameters relating to the running process: The host does not maintain any information about what the plug-in did with the last block of data it processed.
>
> From the host application's point of view, a VST Plug-In is a black box with an arbitrary number of inputs, outputs (MIDI or Audio), and associated parameters. The host needs no implicit knowledge of the plug-in's process to be able to use it. The plug-in process can use whatever parameters it wishes, internally to the process, but depending on the capabilities of the host, it can allow the changes to user parameters to be automated by the host.

cPlay supports use of a single VST plugin (up to version 2.4.2, 2-in/max 8-out) in 32 bit float precision. Plugin is processed after resampling and volume (if used). For best results, use Tiny or Small buffer setting. ASIO performance is maintained for 2-in/2-out plugins (stereo output). For multi-channel output, best ASIO performance is achieved when latency is 64 samples or less.

Settings:

- Bypass button becomes available when a plugin is active. On stereo output DSP output is passed through. On multi-output, only the first 2 mapped ASIO outputs are used (for left & right). On mono output, DSP output is summed & averaged for selected mono ASIO output channel.
- Checking VST option enables plugin to be used in safe mode which limits control. VST editor is accessed through a prompt, bypass is not allowed, and autoplay is not done at startup.

- Plugin setup is done using the VST button:



  VST settings only take effect on a restart. Select plugin (.DLL file) using "VST Plugin" button. "Delete" button clears all VST settings.

  For stereo output, the default Left/Right ASIO setting is used. For multiple outputs, VST to ASIO output setup allows for each ASIO channel to be enabled. This removes unnecessary ASIO overheads when using less plugin outputs, e.g. of plugin's 8-outs only 6 are used.

- After a restart, plugin becomes operational (see Diagnostics for details). cPlay's main window enables VST button to access the plugin's Editor. Should a plugin not offer an Editor, VST button remains disabled.

Popular plugins:

- Allocator is formally called the Frequency Allocator and allows one to eliminate speaker x-overs (and associated nasties). Instead, digital processing is used: stereo is converted into pairs of x-over frequencies (low L+R, mid L+R, etc.) whose amplified analogue signal is then fed directly to a speaker.

- Room Correction is useful. Vendors are limited and expensive.

- Chaining multiple effects can be achieved using the free Effects Chainer (2-in/2-out) or the excellent Console plugin. Here's a multi-effect chain of *EQ > Allocator > Spectrum Analyser (to verify Allocator's outputs)*: